Auto-Create and Log Time to Jira from Git Commits

Step 1: Setup Environment

- Create a `.env` file in your project root or shared scripts folder with the following:

  JIRA_BASE_URL=https://safaricom.atlassian.net

  JIRA_EMAIL=your-email@example.com

  JIRA_API_TOKEN=your_api_token_here

  JIRA_PROJECT_KEY=PN

  JIRA_ACCOUNT_ID=your_jira_account_id

Step 2: Get Your Jira Account ID

- Run the following command to fetch your Jira account ID:

  curl -u your-email@example.com:your_api_token \

    -H "Accept: application/json" \

    "https://safaricom.atlassian.net/rest/api/3/myself"

- Copy the `accountId` from the response and update your `.env`

Step 3: Install Dependencies

- Install required Python packages:

  pip install requests python-dotenv

Step 4: Create the Hook Script

- Save the `jira_commit_handler.py` script in your `.git/hooks/` directory.

- Make sure the script is executable:

  chmod +x .git/hooks/jira_commit_handler.py

Auto-Create and Log Time to Jira from Git Commits

Step 5: Add Post-Commit Hook

- Create a `.git/hooks/post-commit` file with:

  #!/bin/bash

  python3 .git/hooks/jira_commit_handler.py

Step 6: Extract and Log Time

- In your commit message, include `#<minutes>` to log time automatically.

- Example: git commit -m "Fix login bug #30"

- This will log 30 minutes to the created Jira task.

Step 7: Auto-Assign the Issue

- The script uses your `JIRA_ACCOUNT_ID` from the `.env` file to assign the issue to you.

- If missing, the issue won't be assigned automatically.

Step 8: View Created Issue and Logged Time

- After commit, the terminal will show the Jira issue key (e.g., PN-123).

- Open in browser: https://safaricom.atlassian.net/browse/PN-123

- View time in the Time Tracking panel or Worklog tab.